

## Pengenalan Ekspresi Wajah Dari *Cross Dataset* Menggunakan *Convolutional Neural Network (CNN)*

Prana Pramudita Kusdiananggalih<sup>1</sup>, Ema Rachmawati<sup>2</sup>, Risnandar<sup>3</sup>

<sup>1,2,3</sup>Fakultas Informatika, Universitas Telkom, Bandung

<sup>1</sup>pranapramudita@students.telkomuniversity.ac.id, <sup>2</sup>emarachmawati@telkomuniversity.ac.id,

<sup>3</sup>risnandartelyu@telkomuniversity.ac.id

---

### Abstrak

Ekspresi wajah manusia dapat menggambarkan emosi seseorang, dengan mengetahui ekspresi wajah manusia, proses pengenalan emosi manusia akan terbantu. Pengenalan ekspresi wajah manusia memiliki banyak manfaat dalam berbagai bidang. Contoh bidang pemanfaatannya dari sistem ini adalah mengenali kepuasan individu terhadap suatu layanan. Salah satu metode yang cukup terkenal saat ini untuk sistem pengenalan ekspresi wajah (*Facial Expression Recognition*) adalah *Convolutional Neural Network (CNN)*. Pada penelitian ini akan dibangun arsitektur *CNN* yang memiliki 8 *layer convolution*, dengan *depth* sebanyak 32 *layer*. Hampir seluruh penelitian pengenalan ekspresi wajah sampai saat ini menggunakan *dataset* ras non-Indonesia. Oleh karena itu, penulis melakukan analisis terhadap *dataset* ras non-Indonesia dengan *dataset* ras Indonesia menggunakan teknik *cross dataset*. Pada sistem ini dilakukan perbandingan antara *self built CNN* dengan arsitektur *CNN* populer lainnya. Hasil yang didapatkan dari penelitian ini adalah akurasi data uji sebesar 91,29%, *sensitivity* atau *recall* atau *True Positive Rate (TPR)* sebesar 91,29%, *precision* atau *Positive Predictive Value (PPV)* sebesar 91,29%, dan *overall accuracy* sebesar 97,51%. Oleh karena itu, dengan nilai *recall* dan *precision* yang tinggi, artinya kelas pada data uji ditangani dengan sempurna oleh model yang dibangun.

**Kata kunci :** facial expression recognition (FER), convolutional neural network (CNN), cross dataset.

---

### Abstract

Human facial expressions can describe a person's emotions, by knowing human facial expressions, the process of recognizing human emotions will be helped. For example is to recognize individual satisfaction of a service. One method that is well-known today for facial expression recognition systems is the Convolutional Neural Network (CNN). In this study, a CNN architecture will be built which has 8 convolution layers, with a depth of 32 layers. Almost all research on facial expression recognition has used datasets of non-Indonesian races. Therefore, the authors conducted an analysis of the non-Indonesian racial dataset with the Indonesian race dataset using the cross dataset technique. In this system the self-built CNN is compared with other popular CNN architectures. The results obtained from this study are the accuracy of the test data by 91.29%, sensitivity or recall or True Positive Rate (TPR) by 91.29%, precision or Positive Predictive Value (PPV) by 91.29%, and overall accuracy by 97.51%. Therefore, with a high recall value and precision, it means that the classes in the test data are handled perfectly by the model built.

**Keywords :** facial expression recognition (FER), convolutional neural network (CNN), cross dataset.

---

### 1. Pendahuluan

#### Latar Belakang

Pemrosesan wajah (*face processing*) merupakan salah satu teknologi dalam bidang *computer vision* yang paling berkembang saat ini. Salah satunya yaitu Pengenalan Ekspresi Wajah (*Facial Expression Recognition*). Pengenalan ekspresi wajah sering dikaitkan dengan emosi. Pada bidang marketing, kepuasan pelanggan terhadap pelayanan penting untuk diperhatikan oleh perusahaan. Salah satu indikator yang dapat menunjukkan kepuasan pelanggan terhadap pelayanan adalah ekspresi wajah atau emosi pelanggan. Dengan adanya Pengenalan Ekspresi Wajah, perusahaan telah dibukakan peluang untuk meningkatkan tingkat loyalitas pelanggan terhadap perusahaan. Oleh, karena itu penulis tertarik untuk membangun sistem pengenalan ekspresi wajah.

Sejauh ini, sudah banyak peneliti yang melakukan penelitian mengenai Pengenalan Ekspresi Wajah (*Facial Expression Recognition*) hingga tahun 2019 [1]. Dari pendekatan konvensional hingga pendekatan *Deep Learning*. *Dataset* yang digunakan juga bermacam-macam antara lain JAFFE, CK+, MMI, FERA, FER2013, BU-3DFE, Multi-PIE, SFEW, Oulu-Casia. Salah satu penelitian menggunakan cara konvensional yaitu metode Gabor + *Support Vector Machine (SVM)* diterapkan pada *dataset* JAFFE menghasilkan tingkat akurasi sebesar 80,95% [1], [2]. Untuk salah satu penelitian menggunakan *Deep Learning* salah satunya yaitu CNN, pada

*dataset* FER2013 hanya mencapai akurasi sebesar 72,10% [1],[3]. Lalu ada dua penelitian yang menggunakan *dataset* yang sama yaitu FER2013 menggunakan metode CNN (DeeperCNN) dan CubicSVM+HOG mendapat akurasi sebesar 61,10% dan 57,17% [1]. Sedangkan menggunakan metode yang sama yaitu CNN pada *dataset* lain antara lain, CK+ mencapai 98,62%, MMI mencapai 63,40%, FERA mencapai 76,70%, Multi-PIE mencapai 94,70%, SFEW mencapai 47,70%, dan Oulu-CASIA mencapai 58,18% [1]. Hasil akurasi pada tiap *dataset* dan masing-masing metode yang digunakan memiliki akurasi yang berbeda-beda, tentunya banyak sekali parameter-parameter atau faktor-faktor yang dapat mempengaruhi tingkat akurasi. Salah satunya yaitu masing-masing *dataset* memiliki tantangannya tersendiri.

Hampir seluruh penelitian yang dilakukan saat ini, menggunakan *dataset* yang berisi data orang-orang non-Indonesia sebagai data latih maupun data uji [1]. Hal ini memberikan peneliti ide untuk menerapkan metode *Cross Dataset* di mana data latih dan data uji menggunakan *dataset* yang berbeda [3], FER2013 sebagai data latih dan Data-data orang Indonesia (yang peneliti kumpulkan sendiri) sebagai data uji.

Pada suatu penelitian yang dilakukan pada tahun 2016 oleh Mollahosseini, Ali Chan, David Mahoor, dan Mohammad H [4]. Telah dilakukan uji coba terhadap beberapa *dataset* salah satunya terdapat FER2013 [4]. Pada penelitian tersebut, dilakukan perbandingan antara *proposed architecture CNN* dengan arsitektur populer yaitu AlexNet. Untuk *dataset* FER2013 menggunakan *proposed architecture CNN* menghasilkan akurasi sebesar 66,4%, sedangkan AlexNet menghasilkan akurasi sebesar 61,1% [4]. Hasil lain pada *dataset* MMI dihasilkan akurasi sebesar 77,9% pada *proposed architecture CNN*, sedangkan AlexNet menghasilkan akurasi sebesar 56,0% [4]. Hal ini menunjukkan bahwa arsitektur yang dibangun sendiri memungkinkan lebih baik daripada *DeeperCNN* seperti AlexNet untuk beberapa *dataset*.

Pengenalan ekspresi wajah sudah banyak dilakukannya sampai tahun 2019 [1], akan tetapi untuk penerapan pengenalan ekspresi wajah menggunakan *dataset* orang-orang Indonesia, masih sedikit. Oleh karena itu, peneliti mengusulkan untuk membangun sistem pengenalan ekspresi wajah secara *Cross Dataset* menggunakan metode *Convolutional Neural Network* (CNN).

### Topik dan Batasannya

Definisi beberapa dari kata kunci yang ada pada abstrak :

1. Facial Expression Recognition (FER)  
Atau dalam Bahasa Indonesia artinya pengenalan ekspresi wajah manusia adalah sistem yang dapat mengklasifikasi sebuah citra wajah manusia kedalam kategori emosi seperti bahagia, netral, sedih, marah, terkejut, jijik, dan takut.
2. Convolutional Neural Network (CNN)  
Merupakan metode *Deep Learning* yang sangat populer dan banyak digunakan hingga saat ini. Metode ini sangat populer karena kecepatannya dalam melakukan *training data*, tersedia banyak library (contoh : *KERAS*), tidak perlu membuat fitur ekstraksi secara manual
3. *Cross Dataset*  
*Cross dataset* adalah melakukan silang antara data latih dan data uji antar *dataset*. Sebagai contoh dataset CK+, FER2013 dan NovaEmotions. Masing-masing *dataset* memiliki data latih dan data uji. Dan masing-masing data latih dan data uji akan dilakukan silang, sehingga menghasilkan tingkat akurasi masing-masing. Berikut merupakan contoh *cross dataset* yang digunakan dapat dilihat pada tabel 1.1 :

**Tabel 1.1** Contoh tabel *cross dataset* [3]

Train \ Test	CK+	FER2013	NovaEmotions
CK+	98.62%	69.3%	67.2%
FER2013	92.0%	72.1%	78.0%
NovaEmotions	93.75%	71.8%	81.3%

Berikut terdapat beberapa Batasan-batasan masalah :

1. *Dataset* yang digunakan adalah FER2013 dan *Dataset* yang dibangun sendiri
2. *Dataset* yang dibangun sendiri diberi nama "*Self built dataset*"
3. *Dataset* yang dibangun sendiri diambil dari internet lalu dikumpulkan dan dilabeli secara manual
4. *Dataset* FER-2013 yang digunakan berjumlah 35.685 (sebagai data latih dan data validasi)  
Jumlah *dataset* FER-2013 secara keseluruhan yaitu 35.685, namun yang akan digunakan sebagai data latih berjumlah 28.709 dan sisanya berjumlah 6.976 digunakan sebagai data validasi
5. Kedua *dataset* yang digunakan berisi data dengan informasi piksel (0-255) dengan ukuran matriks 48x48 (bukan data berbentuk format citra JPEG, PNG dsb.) yang disimpan dalam format CSV
6. *Dataset* yang dibangun sendiri berjumlah terbatas, yaitu 379 (sebagai data uji)

Karena keterbatasan sumber daya dan waktu peneliti hanya sanggup mengumpulkan kurang lebih 400 citra berisi ekspresi wajah orang Indonesia sebagai data uji. Proses pengumpulan data dilakukan secara manual yaitu dibagi menjadi berbagai tahapan antara lain pengumpulan referensi media yaitu foto wajah orang Indonesia di Internet, Youtube, Film. Lalu dilakukan *crop* scene/frame dengan ukuran 48x48 (1:1 rasio) berisikan ekspresi wajah orang Indonesia secara manual. Setelah itu disimpan dalam format JPEG, lalu dimasukkan ke dalam masing-masing folder kategori emosi secara manual (sebagai labeling). Peneliti membuat code python yang akan melakukan ekstraksi citra format JPEG menjadi sebuah barisan angka dalam format 0-255 dan akan memberikan label berdasarkan letak file citra dalam folder yang telah dilabel secara manual lalu disimpan ke dalam format CSV. Selain itu, terdapat beberapa ekspresi wajah yang sulit didapat yaitu ekspresi 'jijik' atau *disgust* sangat sulit didapatkan di Internet, Film, media. Oleh karena itu, jumlah *dataset* 'jijik' atau *disgust* sangat terbatas.

7. *Dataset* memiliki format warna abu-abu (*Grayscale*)  
*Dataset* FER-2013 memang memiliki format *grayscale*, oleh karena itu, pada *dataset* yang dibangun sendiri oleh peneliti, harus dikonversi dari RGB menjadi *grayscale* menggunakan metode *weighted method*
8. *Dataset* memiliki ukuran 48x48 yaitu rasio 1:1  
Sehingga citra gambar hanya merupakan citra wajah, sehingga tidak perlu melakukan *Face Detection* pada penelitian ini
9. Ekspresi wajah dikategorikan menjadi tujuh klasifikasi  
Yaitu antara lain bahagia, netral, sedih, marah, terkejut, jijik, dan takut
10. *Dataset* FER2013 memiliki banyak anomali  
Berdasarkan penelitian FER pada *dataset* FER2013, akurasi yang didapat sekitar 57.17-61.10% hal ini dikarenakan *dataset* memiliki tantangan kondisi tersendiri. Subjek memiliki ekspresi spontan yang di bawah berada pada keadaan yang unik. Bahkan terdapat wajah citra kartun pada *dataset*. [1]
11. Batas usia dan jenis kelamin objek manusia  
Untuk *dataset* FER2013 terdapat beragam usia dan jenis kelamin manusia antara lain usia bayi (0-1 tahun), anak-anak (2-10 tahun), remaja (11-19 tahun), dewasa (20-60 tahun) serta jenis kelamin laki-laki dan perempuan. Untuk *self built dataset* antara lain anak-anak (2-10 tahun), remaja (11-19 tahun), dewasa (20-60 tahun) dengan jenis kelamin laki-laki dan perempuan.

## Tujuan

Tujuan dari Tugas Akhir (TA) ini adalah membangun sistem pengenalan ekspresi wajah dari *cross dataset* menggunakan metode *Convolutional Neural Network* (CNN).

## Organisasi Tulisan

Laporan tugas akhir ini terdiri dari lima bagian. Pada bagian pertama menjelaskan pendahuluan, pada bagian kedua menjelaskan studi terkait, pada bagian ketiga menjelaskan sistem yang dibangun, pada bagian keempat menjelaskan evaluasi dari sistem yang dibangun, dan pada bagian kelima menjelaskan kesimpulan yang diambil dari tugas akhir ini.

## 2. Studi Terkait

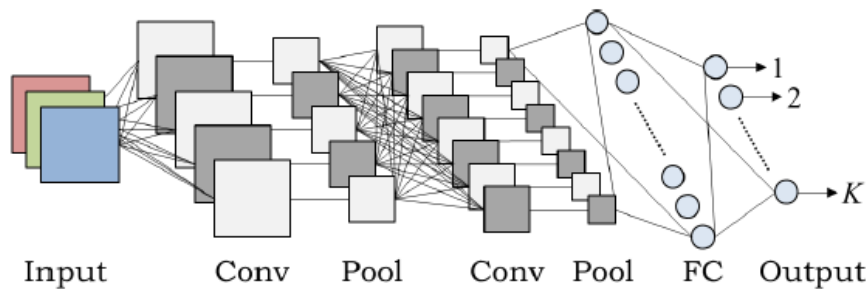
### 2.1 Pengenalan Ekspresi Wajah

Ekspresi wajah manusia merupakan sarana komunikasi non-verbal utama. Pada tahun 1975 penelitian menunjukkan bahwa 55% pesan yang berkaitan dengan perasaan dan atitit adalah ekspresi wajah. 7% kata-kata yang diucapkan. Dan sisanya paralinguistik [5]. Dengan perkembangan teknologi kecerdasan buatan yang pesat, pengenalan ekspresi wajah secara otomatis telah dipelajari secara intensif dalam beberapa tahun akhir pada 2019. *Facial Expression Recognition (FER)* telah menerima perhatian banyak dari bidang psikologi, *computer vision*, dan *pattern recognition*. FER memiliki penerapan yang luas dalam berbagai bidang, misalnya *computer interaction*, *virtual reality*, *augmented reality*, *advanced driver assistance system*, dan pendidikan. Pendekatan untuk pengenalan ekspresi wajah dibagi menjadi dua pendekatan yaitu Pendekatan Konvensional (*Conventional FER Approaches*) dan Pendekatan *Deep Learning (Deep Learning-Based FER Approaches)* [1].

### 2.2 Convolutional Neural Network (CNN)

*Convolutional Neural Network (CNN)* adalah salah satu algoritma deep learning yang dirancang untuk mengolah data dalam berbagai bentuk dimensi, seperti satu dimensi, dua dimensi, dan tiga dimensi. CNN satu dimensi bergerak hanya satu arah. Input outputnya dua dimensi. Biasa digunakan untuk data time-series. CNN dua dimensi bergerak dengan dua arah. *Input* dan *output*-nya tiga dimensi. Biasa digunakan untuk data citra. Sedangkan CNN tiga dimensi bergerak dengan tiga arah. Input dan outputnya empat dimensi. Biasa digunakan pada data citra tiga dimensi (3D). Fungsi utama dari algoritma CNN ini adalah mengklasifikasi masing-masing data latih dengan tujuan untuk mengelompokkan suatu data dengan data yang sebelumnya.

CNN memiliki variasi arsitektur yang berbeda-beda. Dapat terdiri dari *Convolutional Layer*, *Pooling Layer*, dan *Fully-Connected Layer*. Dan bisa juga terdiri dari arsitektur yang lebih detil yaitu *Input*, *Convolutional Layer*, *ReLU Layer*, *Pooling Layer*, *Flatten Layer*, *Fully Connected Layer* dan *Softmax Layer*.



Gambar 2.1 Contoh arsitektur CNN [6]

Secara umum *Convolutional Neural Network (CNN)* dibagi menjadi dua tahapan yaitu:

1. Ekstraksi Fitur, didalamnya terdapat beberapa lapisan dan tiap lapisan saling terhubung. Lapisan pertama adalah *Convolution Layer + ReLU Layer* dan lapisan kedua adalah *Pooling Layer*. Setiap lapisan memiliki masing-masing fungsi. Tahap ini memproses input citra menjadi output berupa vektor untuk diolah pada lapisan berikutnya.

2. Klasifikasi, didalamnya terdapat lapisan-lapisan yang saling terhubung (*Fully connected layer*). Tahap ini menerima hasil input, hasil output pada lapisan ekstraksi fitur. Pada lapisan *Flatten layer* terjadi proses perubahan matriks dua dimensi menjadi satu dimensi. Dan terakhir yaitu lapisan *Softmax Layer* akan mengubah hasil *neural network* menjadi probabilitas terhadap kelas yang dideteksi.

Kesimpulannya CNN merupakan metode untuk mentransformasikan nilai *pixel* pada citra menjadi nilai terhadap kelas untuk diklasifikasi.

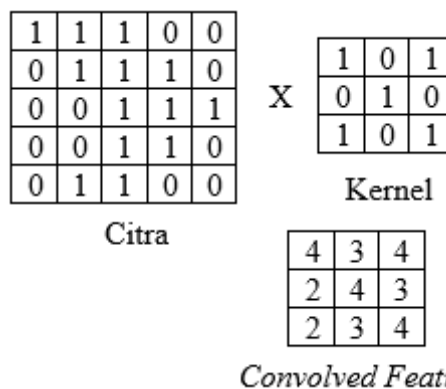
2.2.1 Convolution Layer

*Convolution Layer* memiliki beberapa parameter, yaitu ukuran *kernel*, *skipping factors* dan *connection table*. *Kernel* dalam CNN selalu bergeser pada input, sedangkan *skipping factors* merupakan jumlah pixel yang bergeser pada *kernel*. Ukuran dari *output* pada *map* adalah sebagai berikut :

$$M_x^n = \frac{M_x^{n-1} - K_x^n}{S_x^n + 1} + 1; M_y^n = \frac{M_y^{n-1} - K_y^n}{S_y^n + 1} + 1;$$

Di mana:

- $M_x^n, M_y^n$  : Ukuran feature maps
- $S_x, S_y$  : Skipping factors
- $K_x, K_y$  : Ukuran *kernel*
- n : Letak *layer* saat proses



Gambar 2.2 Contoh Konvolusi citra ukuran 5x5x1 dengan 3x3x1 kernel.

### 2.2.2 Pooling Layer

*Pooling Layer* adalah proses pengurangan ukuran citra, mengurangi jumlah parameter dan jumlah perhitungan dalam jaringan serta melakukan pencegahan *overfitting*. Pooling layer terbagi menjadi dua yaitu *max pooling* dan *average pooling*. *Max pooling* yaitu mengambil nilai terbesar dari area yang dipilih, sedangkan *Average pooling* mengambil nilai rata-ratanya. Persamaan dari *max pooling* sebagai berikut :

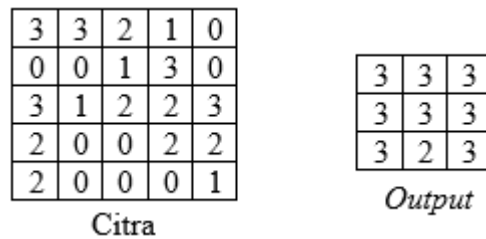
$$a_j = \max_{N \times N} a_i^{n \times n} \mu(n, n)$$

Di mana:

$a_i$  : Value dari *pooling map*

$a_j$  : Value dari *input map*

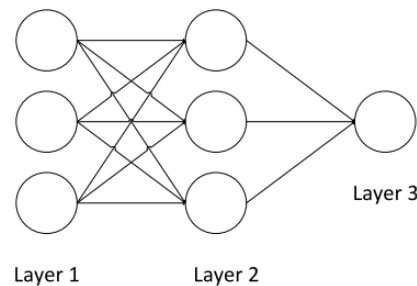
$\mu(n, n)$  : *Window function*



Gambar 2.3 Contoh *Max pooling layer* 3x3 pada matriks 5x5

### 2.2.3 Fully Connected Layer

*Fully Connected Layer* adalah layer yang menghubungkan setiap neuron dari layer ke layer lainnya. Berikut contoh dari *Fully Connected Layer* :



Gambar 2.4 Ilustrasi *Fully Connected Layer* [7]

### 2.2.4 Rectified Linear Unit

*Rectified Linear Unit (RELU)* merupakan salah satu fungsi aktivasi, Kelebihan dari penggunaan RELU adalah fungsi ini mampu memproses data yang besar dengan cepat. Berikut bentuk fungsinya :

$$f = \max(0, x)$$

### 2.2.5 Softmax

*Softmax* merupakan salah satu fungsi aktivasi yang memiliki persamaan sebagai berikut :

$$f_i(x) = \frac{e^{x_i}}{\sum_{j=1}^k e^{x_j}}$$

$$\text{where } i = 1, 2, 3, \dots, k$$

$$0 \leq f_i(x) \leq 1$$

Di mana:

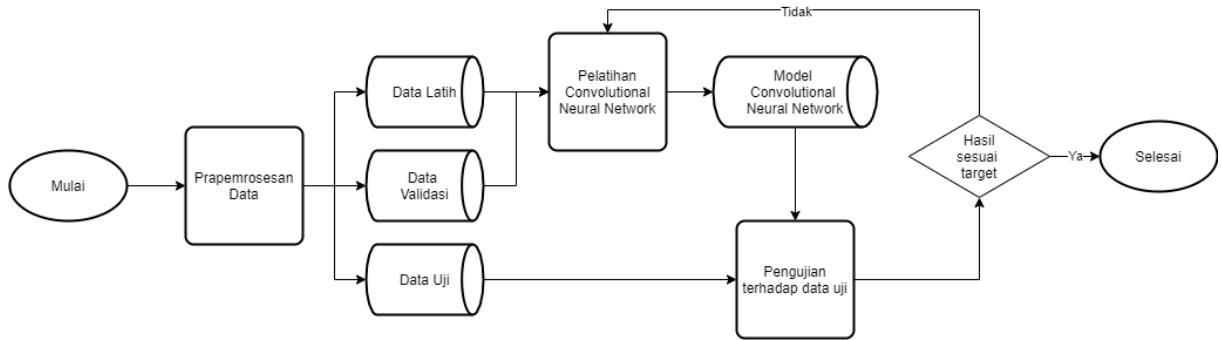
k : Jumlah input

x : Vektor input

## 3. Sistem yang Dibangun

### 3.1 Gambaran Umum Sistem

Sistem pengenalan ekspresi wajah ini menggunakan *Convolutional Neural Network*. Dalam melakukan pengenalan ekspresi wajah ini, terdapat dua proses yaitu prapemrosesan data dan melakukan pelatihan serta pengujian *Convolutional Neural Network*. Berikut ini adalah diagram flowchart dari sistem pengenalan ekspresi wajah menggunakan *Convolutional Neural Network* :



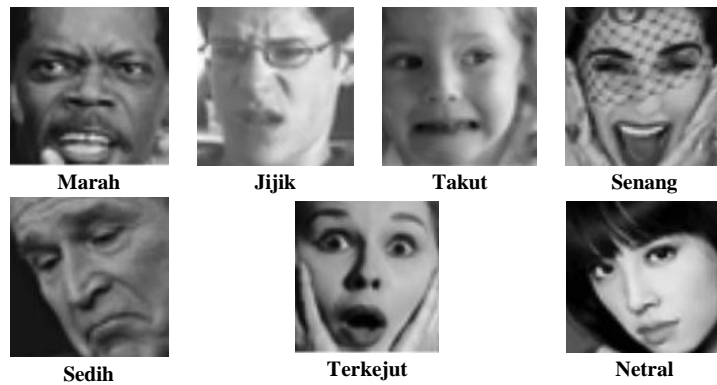
Gambar 3.1 Diagram Flowchart Rancangan Sistem

3.2 Dataset

Dataset yang digunakan pada sistem ini ada dua yaitu :

1. FER2013

Dataset yang digunakan pada sistem ini adalah FER2013 [1]. Ukuran citra adalah matriks 48x48 dan memiliki format warna abu-abu (*grayscale*). Pada dataset terdapat tujuh label klasifikasi, antara lain marah, jijik, takut, senang, sedih, kaget, dan netral. Format dataset berupa file berformat .csv. Di dalamnya terdapat dua kolom yaitu “emotion” dan “pixels”. Kolom *emotion* memiliki nilai antara 0-6 (0 = marah, 1 = jijik, 2 = takut, 3 = senang, 4 = sedih, 5 = kaget, 6 = netral). Sedangkan kolom *pixels* berisi sebuah matriks berukuran 48x48 yang memiliki nilai warna dalam format 0-255. Berikut merupakan contoh Citra yang ada pada dataset FER-2013 :



Gambar 3.2 Marah, Jijik, Takut, Senang, Sedih, Kaget, Netral (Kiri-Kanan)  
Berikut merupakan informasi jumlah citra pada tiap kelas dataset FER2013 :

Tabel 3.1 Jumlah citra tiap kelas dataset FER2013 (Data latih)

FER2013 (Data Latih)	
Kelas (Emosi)	Jumlah
Marah	3995
Jijik	436
Takut	4097
Senang	7215
Sedih	4830
Kaget	3171
Netral	4965
Total	28709

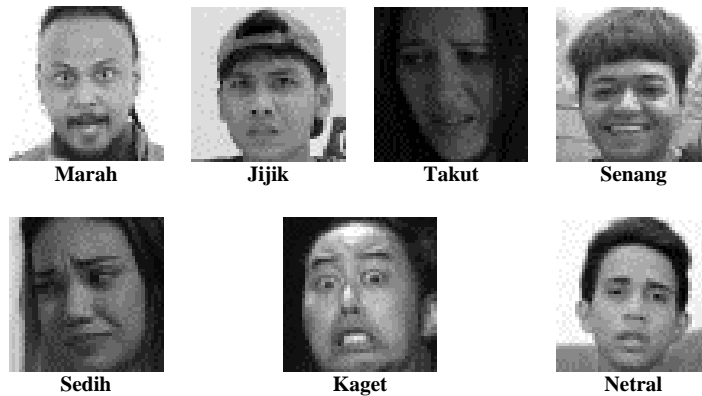
Tabel 3.2 Jumlah citra tiap kelas dataset FER2013 (Data validasi)

FER2013 (Data Validasi)	
Kelas (Emosi)	Jumlah
Marah	958
Jijik	111
Takut	1024
Senang	1774
Sedih	1247

Kaget	831
Netral	1233
Total	7178

2. *Self Built Dataset*

Merupakan *dataset* yang dibangun sendiri. *Dataset* ini adalah *dataset* yang peneliti kumpulkan dari internet seperti media Youtube, Film. *Dataset* berisi citra wajah orang-orang Indonesia. *Dataset* telah dicrop menjadi ukuran 48x48 dan memiliki *RGB* yang akan dikonversikan ke dalam *grayscale* pada tahap prapemrosesan. Serta, kumpulan *dataset* akan peneliti labeli secara manual. Format citra dibuat sedemikian rupa menyerupai *dataset* data latih FER2013. Berikut beberapa contoh *dataset* yang dibangun sendiri yang akan digunakan sebagai data uji :



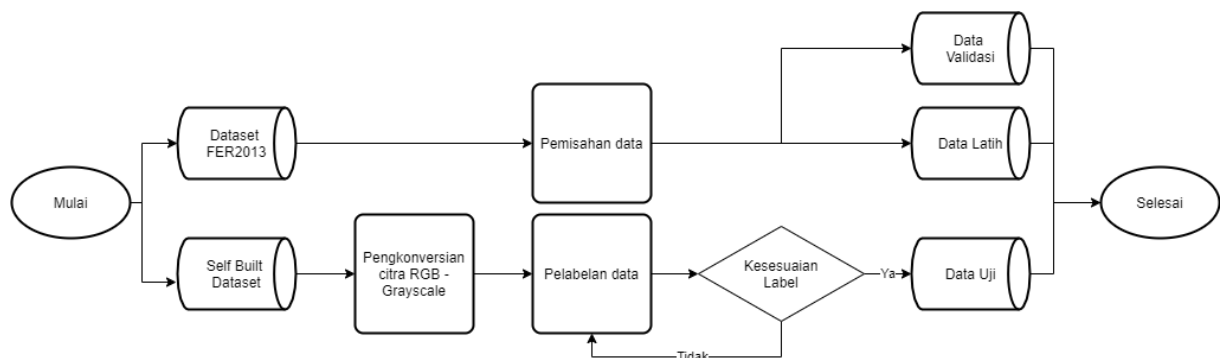
**Gambar 3.3** Marah, Jijik, Takut, Senang, Sedih, Kaget, Netral (Kiri-Kanan)  
Berikut merupakan jumlah citra pada tiap kelas *Self Built Dataset* :

**Tabel 3.3** Jumlah citra tiap kelas *Self Built Dataset* (Data uji)

<i>Self Built Dataset</i> (Data Uji)	
Kelas (Emosi)	Jumlah
Marah	28
Jijik	34
Takut	23
Senang	133
Sedih	46
Kaget	26
Netral	89
Total	379

3.3 Prapemrosesan Data

Prapemrosesan Data ini perlu dilakukan untuk mengkondisikan *dataset* agar dapat diproses di tahap berikutnya. Tahapan ini dapat dilihat pada gambar berikut :



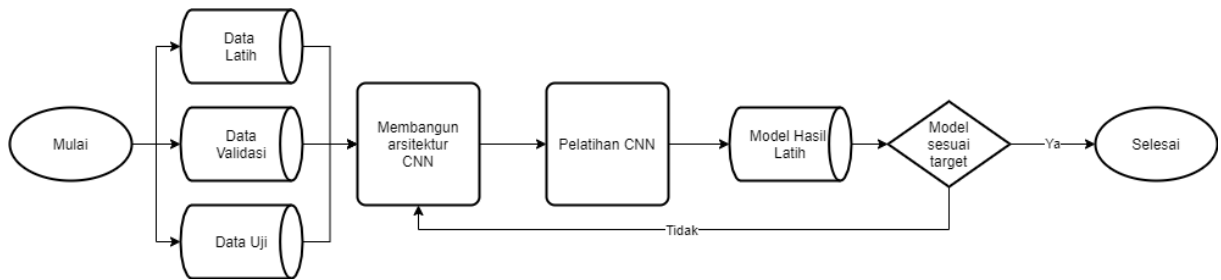
**Gambar 3.4** Diagram Flowchart Prapemrosesan Data

Dataset FER2013 terdiri atas 35.887 data. Yang terdiri atas tiga label yaitu “Training”, “PublicTest”, dan “PrivateTest” masing-masing berjumlah 28.709, 3.589, dan 3.589. Namun, pada rancangan sistem ini hanya menggunakan satu label yaitu “Training” sebagai data latih.

Untuk *Self Built Dataset*, dataset yang dibangun akan dikonversi kedalam format *grayscale* menggunakan metode *weighted method* karena metode ini merupakan metode yang banyak digunakan untuk mengkonversi citra RGB ke *grayscale*. Setelah itu dilakukan pelabelan data kedalam tujuh emosi seperti pada dataset FER-2013, lalu dataset ini akan digunakan sebagai data uji pada sistem ini.

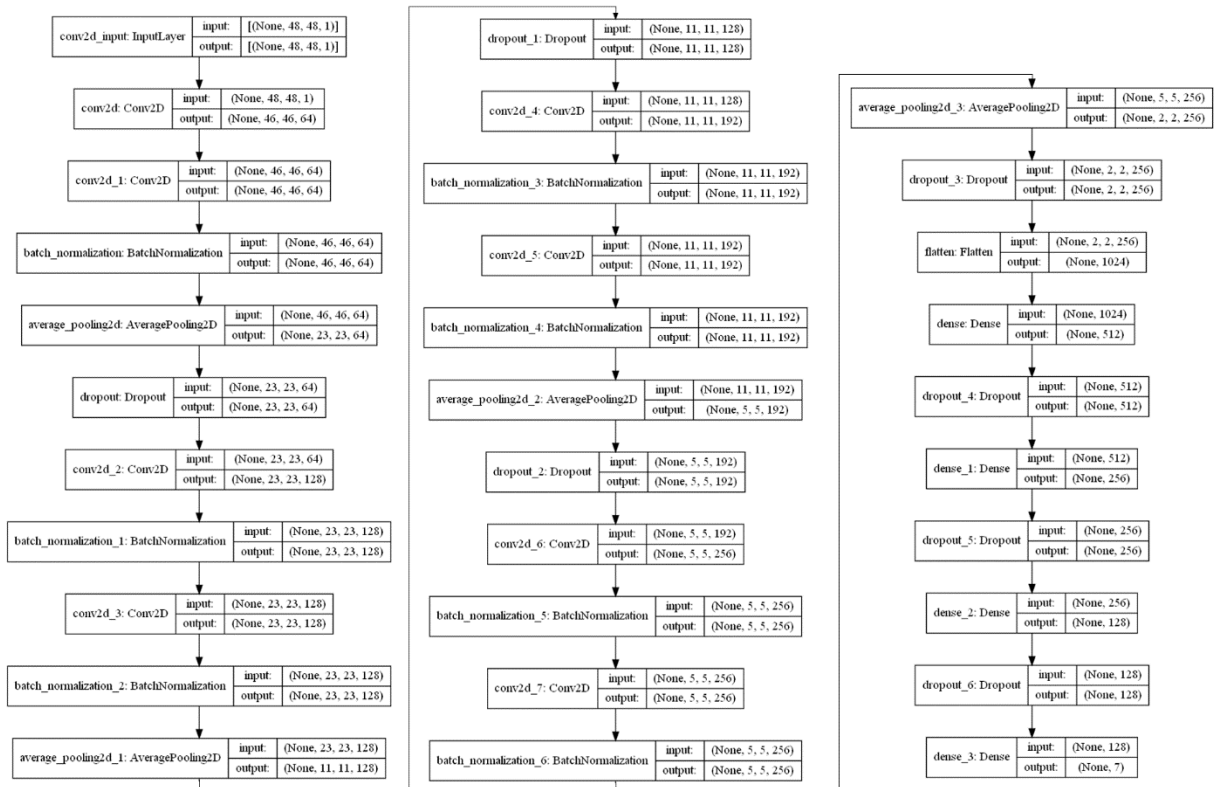
### 3.4 Pelatihan Convolution Neural Network

Pelatihan *Convolution Neural Network* dilakukan untuk membuat model yang akan dilakukan pelatihan untuk diuji performasinya. Tahapan ini dapat dilihat pada gambar berikut :



Gambar 3.5 Diagram Flowchart Pelatihan CNN

Pada proses pelatihan CNN ini memiliki tujuan yaitu untuk membangun model CNN menggunakan arsitektur CNN yang akan dibangun sendiri. Arsitektur CNN memiliki 8 layer konvolusi yang akan digunakan sebagai *training* CNN adalah sebagai berikut :



Gambar 3.6 Arsitektur CNN dengan 8 layer konvolusi

Arsitektur CNN terdiri dari beberapa *Input Layer*, *Convolution Layer*, *Batch Normalization*, *Average Pooling Layer*, *Dropout Layer*, *Fully Connected Layer*. Berikut detail jumlah tiap layer yang digunakan pada arsitektur CNN yang dibangun sendiri :

Tabel 3.4 Layer arsitektur CNN yang dibangun sendiri

Layer	Jumlah
-------	--------



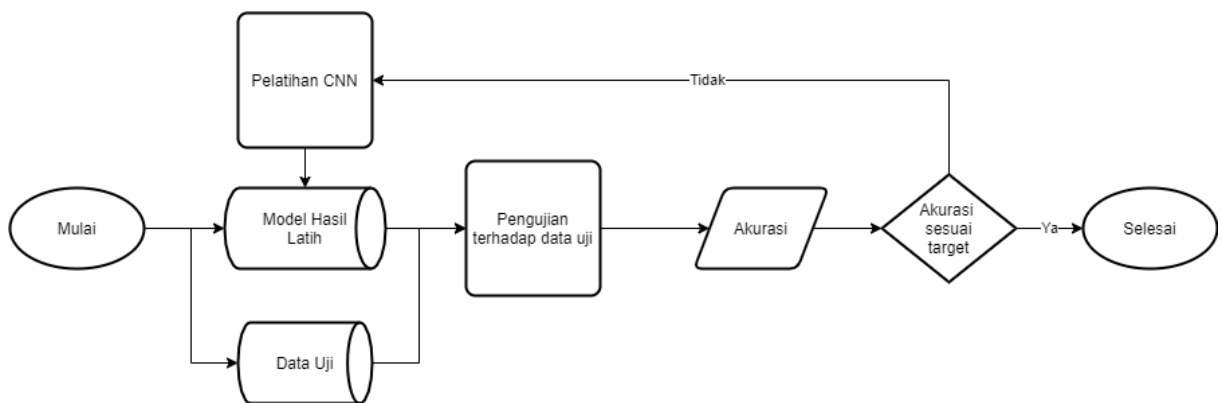
<i>Input Layer (Feature Extraction Convolution Layer)</i>	<i>1 Layer</i>
<i>Convolution Layer</i>	<i>8 Layer</i>
<i>Batch Normalization</i>	<i>7 Layer</i>
<i>Average Pooling Layer</i>	<i>4 Layer</i>
<i>Dropout Layer</i>	<i>7 Layer</i>
<i>Flatten</i>	<i>1 Layer</i>
<i>Dense</i>	<i>4 Layer</i>
<b>Total</b>	<b>32 Layer</b>

*Pooling Layer* yang digunakan adalah *Average Pooling Layer*, yaitu dengan mengambil nilai rata-rata dari bagian layer. Tujuan dari *pooling layer* ini adalah *downsampling* agar mempercepat komputasi. Fungsi aktivasi yang digunakan adalah *ReLU*, aktivasi ini umum digunakan karena sifatnya yang berfungsi lebih baik dari aktivasi lainnya.

Setelah model hasil latih selesai, kita dapat mengujinya pada *Self built dataset* sehingga menghasilkan nilai akurasi dari model yang dibangun.

### 3.5 Pengujian Terhadap Data Uji

Pengujian terhadap data uji dilakukan untuk mengukur performansi sistem yang telah dibuat. Hasil dari pengujian ini adalah berupa nilai akurasi. Tahapan proses ini dapat dilihat pada diagram sebagai berikut :



**Gambar 3.7** Diagram *Flowchart* Pengujian Terhadap Data Uji

## 4. Evaluasi

Dalam penelitian ini akan dilakukan beberapa skenario yang dilakukan yaitu Tuning jumlah lapisan konvolusi dan Tuning *data augmentation*

### 4.1 Hasil Pengujian

Skenario Tuning Jumlah Lapisan Konvolusi

Dilakukan percobaan jumlah lapisan konvolusi CNN terhadap *dataset FER2013* dan *self built dataset*. Hasilnya sebagai berikut :

**Tabel 4.1** Pengujian lapisan konvolusi CNN

<i>Convolution Layer</i>	<i>Train Accuracy (FER2013)</i>	<i>Validation Accuracy (FER2013)</i>	<i>Test Accuracy (Self Built Dataset)</i>
2	0.9973	0.5316	0.4749
4	0.9960	0.5821	0.5330
6	0.9932	0.6247	0.5910
8	0.9796	0.6439	0.6913
10	0.9797	0.6418	0.6887

Pada tabel 4.1, secara keseluruhan nilai akurasi pada data latih sangat tinggi, namun pada data validasi nilainya kecil, dapat disimpulkan bahwa seluruh percobaan mengalami *overfitting*. Oleh karena itu, selanjutnya akan dilakukan skenario tuning *data augmentation*. Dengan menggunakan *data augmentation* dapat mengeneralisasi *dataset* sehingga dapat mengurangi kemungkinan *overfitting*. Namun yang dapat menjadi pertimbangan adalah semakin sedikit lapisan konvolusi maka semakin tinggi kemungkinan *overfitting*. Parameter ini dapat dijadikan pertimbangan penentuan arsitektur CNN yang akan dipilih.

Berikut merupakan hasil perhitungan dari *TPR* (*True Positive Rate/Sensitivity/Recall*), *PPV* (*Positive Predictive Value/Precision*), dan *Accuracy* dari model dengan jumlah konvolusi 2 layer yang diajukan :

**Tabel 4.2** Perhitungan *TPR*, *PPV*, *Accuracy* model 2 konvolusi

<i>Classification</i>	<i>Recall</i>	<i>Precision</i>	<i>Accuracy</i>
<i>Angry</i>	0.2143	0.1579	0.8575
<i>Disgust</i>	0.0294	0.3333	0.9077
<i>Fear</i>	0.2174	0.1613	0.8839
<i>Happy</i>	0.7143	0.6643	0.7731
<i>Sad</i>	0.3478	0.3077	0.8259
<i>Surprise</i>	0.5000	0.7222	0.9525
<i>Neutral</i>	0.4944	0.4681	0.7493
<i>Weighted Average</i>	0.4749	0.4813	0.8113
<b><i>Overall</i></b>	0.4749	0.4749	0.8500

Pada tabel 4.2, secara perhitungan *overall*, *recall* dan *precision* memiliki nilai yang rendah sebesar 47.49%, yang artinya secara keseluruhan kelas tidak ditangani dengan baik oleh model.

Berikut merupakan hasil perhitungan dari *TPR* (*True Positive Rate/Sensitivity/Recall*), *PPV* (*Positive Predictive Value/Precision*), dan *Accuracy* dari model dengan jumlah konvolusi 4 layer yang diajukan :

**Tabel 4.3** Perhitungan *TPR*, *PPV*, *Accuracy* model 4 konvolusi

<i>Classification</i>	<i>Recall</i>	<i>Precision</i>	<i>Accuracy</i>
<i>Angry</i>	0.3214	0.2727	0.8865
<i>Disgust</i>	0.0294	0.500	0.9103
<i>Fear</i>	0.2174	0.2778	0.9182
<i>Happy</i>	0.7895	0.7836	0.8496
<i>Sad</i>	0.4783	0.2973	0.7995
<i>Surprise</i>	0.5385	0.5600	0.9393
<i>Neutral</i>	0.5169	0.4946	0.7625
<i>Weighted Average</i>	0.5330	0.5475	0.8416
<b><i>Overall</i></b>	0.5330	0.5330	0.8666

Pada tabel 4.3, secara perhitungan *overall*, *recall* dan *precision* memiliki nilai yang rendah sebesar 53.30%, yang artinya secara keseluruhan kelas tidak ditangani dengan baik oleh model.

Berikut merupakan hasil perhitungan dari *TPR* (*True Positive Rate/Sensitivity/Recall*), *PPV* (*Positive Predictive Value/Precision*), dan *Accuracy* dari model dengan jumlah konvolusi 6 layer yang diajukan :

**Tabel 4.4** Perhitungan *TPR*, *PPV*, *Accuracy* model 6 konvolusi

<i>Classification</i>	<i>Recall</i>	<i>Precision</i>	<i>Accuracy</i>
<i>Angry</i>	0.3929	0.2391	0.8628
<i>Disgust</i>	0.0000	0.0000	0.9077
<i>Fear</i>	0.3043	0.2333	0.8971
<i>Happy</i>	0.8722	0.9134	0.9261
<i>Sad</i>	0.5217	0.3243	0.8100
<i>Surprise</i>	0.4615	0.6316	0.9446
<i>Neutral</i>	0.6067	0.6585	0.8338
<i>Weighted Average</i>	0.5910	0.5897	0.8835
<b><i>Overall</i></b>	0.5910	0.5910	0.8832

Pada tabel 4.4, secara perhitungan *overall*, *recall* dan *precision* memiliki nilai yang rendah sebesar 59.10%, yang artinya secara keseluruhan kelas tidak ditangani dengan terlalu baik oleh model.

Berikut merupakan hasil perhitungan dari *TPR (True Positive Rate/Sensitivity/Recall)*, *PPV (Positive Predictive Value/Precision)*, dan *Accuracy* dari model dengan jumlah konvolusi 8 layer yang diajukan :

**Tabel 4.5** Perhitungan *TPR, PPV, Accuracy* model **8 konvolusi**

<i>Classification</i>	<i>Recall</i>	<i>Precision</i>	<i>Accuracy</i>
<i>Angry</i>	0.6071	0.3269	0.8786
<i>Disgust</i>	0.0000	0.0000	0.9103
<i>Fear</i>	0.2174	0.2941	0.9208
<i>Happy</i>	0.9173	0.8905	0.9314
<i>Sad</i>	0.5435	0.5319	0.8865
<i>Surprise</i>	0.6154	0.8421	0.9657
<i>Neutral</i>	0.8652	0.7196	0.8892
<i>Weighted Average</i>	0.6913	0.0000	0.9120
<b><i>Overall</i></b>	0.6913	0.6913	0.9118

Pada tabel 4.5, secara perhitungan *overall, recall* dan *precision* memiliki nilai tidak terlalu rendah, yaitu sebesar 69,13%, yang artinya kelas secara keseluruhan sudah terdeteksi cukup baik oleh model.

Berikut merupakan hasil perhitungan dari *TPR (True Positive Rate/Sensitivity/Recall)*, *PPV (Positive Predictive Value/Precision)*, dan *Accuracy* dari model dengan jumlah konvolusi 10 layer yang diajukan :

**Tabel 4.6** Perhitungan *TPR, PPV, Accuracy* model **10 konvolusi**

<i>Classification</i>	<i>Recall</i>	<i>Precision</i>	<i>Accuracy</i>
<i>Angry</i>	0.3929	0.2750	0.8786
<i>Disgust</i>	0.0294	0.5000	0.9103
<i>Fear</i>	0.3913	0.2903	0.9050
<i>Happy</i>	0.9474	0.9130	0.9499
<i>Sad</i>	0.5217	0.5217	0.8839
<i>Surprise</i>	0.8077	0.7000	0.9631
<i>Neutral</i>	0.7753	0.7500	0.8865
<i>Weighted Average</i>	0.6887	0.6907	0.9164
<b><i>Overall</i></b>	0.6887	0.6887	0.9110

Pada tabel 4.6, secara perhitungan *overall, recall* dan *precision* memiliki nilai sebesar 68,87%, yang artinya kelas secara keseluruhan sudah terdeteksi cukup baik oleh model.

#### Skenario Tuning *data augmentation*

*Data augmentation* akan menggeneralisasi *dataset* dengan cara memutar, memiringkan, men-zoom, dll citra untuk mengurangi *overfitting*.

**Tabel 4.7** Pengujian augmentasi data

<i>Convolution Layer</i>	<i>Train Accuracy</i>	<i>Validation Accuracy (FER2013)</i>	<i>Test Accuracy (Self Built Dataset)</i>
2	0.4896	0.4806	0.5910
4	0.5996	0.5829	0.7018
6	0.6530	0.6222	0.7467
8	0.6786	<b>0.6385</b>	<b>0.9129</b>
10	0.6664	0.6315	0.7704

Berdasarkan hasil percobaan pada tabel 4.7, lapisan konvolusi 2 memiliki tingkat akurasi validasi dan akurasi *self built dataset* terendah yaitu 48,06% dan 59,10%. Sedangkan untuk validasi akurasi dan akurasi *self built dataset* tertinggi didapat oleh lapisan konvolusi 8 yaitu sebesar 63,85% dan 91,29%. Jumlah konvolusi 8 dapat dijadikan sebagai pertimbangan parameter jumlah lapisan konvolusi terbaik.

Berikut merupakan hasil perhitungan dari *TPR (True Positive Rate/Sensitivity/Recall)*, *PPV (Positive Predictive Value/Precision)*, dan *Accuracy* dari model dengan jumlah konvolusi 2 layer menggunakan *data augmentation* yang diajukan :

**Tabel 4.8** Perhitungan *TPR, PPV, Accuracy* model **2 konvolusi + data augmentation**

<i>Classification</i>	<i>Recall</i>	<i>Precision</i>	<i>Accuracy</i>
<i>Angry</i>	0.1786	0.1923	0.8839
<i>Disgust</i>	0.0000	0.0000	0.9103
<i>Fear</i>	0.2609	0.3750	0.9288
<i>Happy</i>	0.9098	0.7378	0.8549
<i>Sad</i>	0.2826	0.4062	0.8628
<i>Surprise</i>	0.6538	0.5862	0.9446
<i>Neutral</i>	0.6966	0.5536	0.7968
<i>Weighted Average</i>	0.5910	0.0000	0.8600
<b><i>Overall</i></b>	0.5910	0.5910	0.8832

Pada tabel 4.8, secara perhitungan *overall, recall* dan *precision* memiliki nilai yang kurang tinggi sebesar 59,10%, yang artinya kelas tidak ditangani dengan baik oleh model.

Berikut merupakan hasil perhitungan dari *TPR (True Positive Rate/Sensitivity/Recall)*, *PPV (Positive Predictive Value/Precision)*, dan *Accuracy* dari model dengan jumlah konvolusi 4 layer menggunakan *data augmentation* yang diajukan :

**Tabel 4.9** Perhitungan *TPR, PPV, Accuracy* model **4 konvolusi + data augmentation**

<i>Classification</i>	<i>Recall</i>	<i>Precision</i>	<i>Accuracy</i>
<i>Angry</i>	0.3571	0.3125	0.8945
<i>Disgust</i>	0.0000	0.0000	0.9103
<i>Fear</i>	0.2174	0.8333	0.9499
<i>Happy</i>	0.9398	0.8562	0.9235
<i>Sad</i>	0.6087	0.4667	0.8681
<i>Surprise</i>	0.8077	0.7500	0.9683
<i>Neutral</i>	0.8652	0.7196	0.8892
<i>Weighted Average</i>	0.7018	0.0000	0.9101
<b><i>Overall</i></b>	0.7018	0.7018	0.9148

Pada tabel 4.9, secara perhitungan *overall, recall* dan *precision* memiliki nilai yang kurang tinggi sebesar 70,18%, yang artinya kelas sudah cukup baik ditangani oleh model.

Berikut merupakan hasil perhitungan dari *TPR (True Positive Rate/Sensitivity/Recall)*, *PPV (Positive Predictive Value/Precision)*, dan *Accuracy* dari model dengan jumlah konvolusi 6 layer menggunakan *data augmentation* yang diajukan :

**Tabel 4.10** Perhitungan *TPR, PPV, Accuracy* model **6 konvolusi + data augmentation**

<i>Classification</i>	<i>Recall</i>	<i>Precision</i>	<i>Accuracy</i>
<i>Angry</i>	0.8571	0.4138	0.8997
<i>Disgust</i>	0.0000	0.0000	0.9103
<i>Fear</i>	0.1304	0.3333	0.9314
<i>Happy</i>	0.9699	0.8958	0.9499
<i>Sad</i>	0.6522	0.6250	0.9103
<i>Surprise</i>	0.8077	0.7778	0.9710
<i>Neutral</i>	0.8539	0.8172	0.9208
<i>Weighted Average</i>	0.7467	0.0000	0.9313
<b><i>Overall</i></b>	0.7467	0.7467	0.9276

Pada tabel 4.10, secara perhitungan *overall*, *recall* dan *precision* memiliki nilai yang kurang tinggi sebesar 74,67%, yang artinya kelas sudah cukup baik ditangani oleh model.

Berikut merupakan hasil perhitungan dari *TPR* (*True Positive Rate/Sensitivity/Recall*), *PPV* (*Positive Predictive Value/Precision*), dan *Accuracy* dari model dengan jumlah konvolusi 8 layer menggunakan *data augmentation* yang diajukan :

**Tabel 4.11** Perhitungan *TPR*, *PPV*, *Accuracy* model **8 konvolusi + data augmentation**

<i>Classification</i>	<i>Recall</i>	<i>Precision</i>	<i>Accuracy</i>
<i>Angry</i>	1.0000	0.8235	0.9842
<i>Disgust</i>	0.2353	0.8889	0.9288
<i>Fear</i>	0.8696	0.8333	0.9815
<i>Happy</i>	1.0000	0.9708	0.9894
<i>Sad</i>	0.9348	0.8431	0.9710
<i>Surprise</i>	1.0000	1.0000	1.0000
<i>Neutral</i>	0.9888	0.8980	0.9710
<i>Weighted Average</i>	0.9129	0.9136	0.9773
<b><i>Overall</i></b>	0.9129	0.9129	0.9751

Pada tabel 4.11, secara perhitungan *overall*, *recall* dan *precision* memiliki nilai yang tinggi sebesar 91,29%, yang artinya kelas ditangani dengan sempurna oleh model.

Berikut merupakan hasil perhitungan dari *TPR* (*True Positive Rate/Sensitivity/Recall*), *PPV* (*Positive Predictive Value/Precision*), dan *Accuracy* dari model dengan jumlah konvolusi 10 layer menggunakan *data augmentation* yang diajukan :

**Tabel 4.12** Perhitungan *TPR*, *PPV*, *Accuracy* model **10 konvolusi + data augmentation**

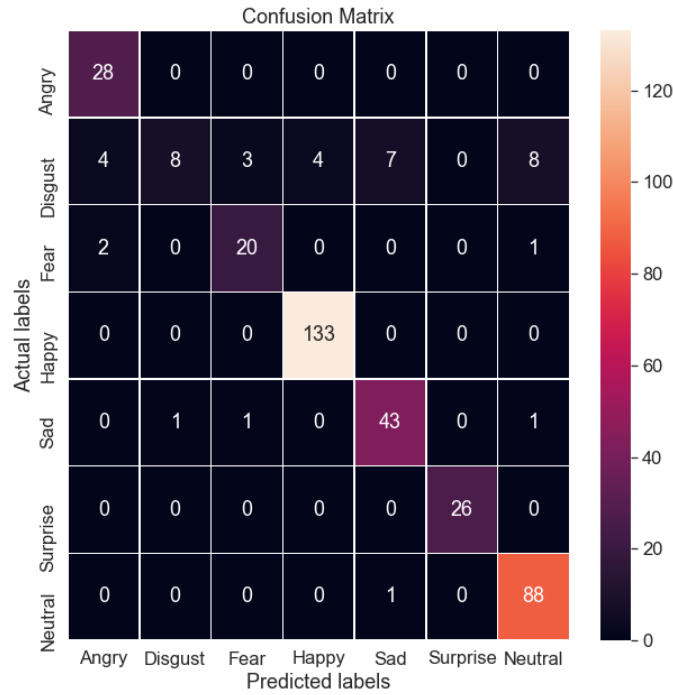
<i>Classification</i>	<i>Recall</i>	<i>Precision</i>	<i>Accuracy</i>
<i>Angry</i>	0.6429	0.4865	0.9235
<i>Disgust</i>	0.1176	0.8000	0.9182
<i>Fear</i>	0.5652	0.5000	0.9393
<i>Happy</i>	0.9549	0.9407	0.9631
<i>Sad</i>	0.6739	0.5636	0.8971
<i>Surprise</i>	0.8077	0.8077	0.9736
<i>Neutral</i>	0.8764	0.8211	0.9261
<i>Weighted Average</i>	0.7704	0.7848	0.9387
<b><i>Overall</i></b>	0.7704	0.7704	0.9344

Pada tabel 4.12, secara perhitungan *overall*, *recall* dan *precision* memiliki nilai yang tinggi sebesar 77,04%, yang artinya kelas sudah cukup baik ditangani oleh model.

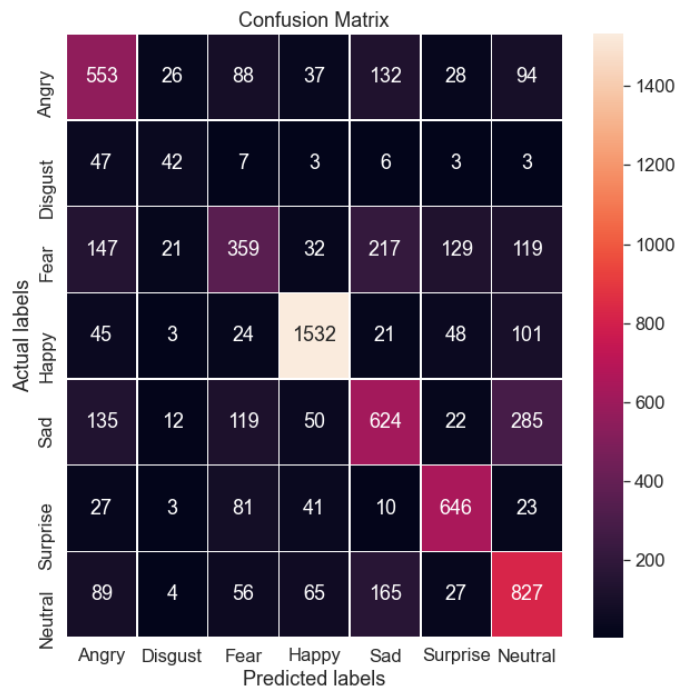
Berdasarkan hasil perbandingan tiap model yang diajukan. Hasil yang terbaik diperoleh oleh model 8 konvolusi menggunakan *data augmentation* dengan *overall recall* dan *precision* sebesar 91,29%, yang artinya kelas secara keseluruhan tertangani dengan sempurna oleh model.

#### 4.2 Analisis Hasil Pengujian

Pada pengujian menggunakan jumlah lapisan konvolusi sebanyak 8 dan *data augmentation*. Didapatkan hasil *confusion matrix* sebagai berikut :



Gambar 4.1. Confusion Matrix Self Built Dataset



Gambar 4.2 Confusion Matrix FER2013

Berikut rumus-rumus yang digunakan untuk menghitung *TPR* (*True Positive Rate/Sensitivity/Recall*), *PPV* (*Positive Predictive Value/Precision*), dan *Accuracy* :

$$Sensitivity = Recall = High Rate = TPR = \frac{TP}{(TP + FN)}$$

$$Precision = PPV = \frac{TP}{(TP + FP)}$$

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)}$$

Di mana:

TPR : *True Positive Rate*

PPV : *Positive Predictive Value*

TP : True Positive  
 TN : True Negative  
 FP : False Positive  
 FN : False Negative

Pada gambar 4.1 dapat dilihat terdapat 28 citra marah terprediksi marah, 8 citra jijik terprediksi jijik, 20 citra takut terprediksi takut, 133 citra senang terprediksi senang, 43 citra sedih terprediksi sedih, 26 citra terkejut terprediksi terkejut, dan 88 citra netral terprediksi netral. Dari hasil pengujian arsitektur CNN 8 lapisan konvolusi menggunakan *data augmentation* didapatkan akurasi data uji sebesar 91,29%. Dengan *precision*, *recall*, dan *accuracy* sebagai berikut :

**Tabel 4.13** TPR (*True Positive Rate/Sensitivity/Recall*), PPV (*Positive Predictive Value/Precision*), dan *Accuracy Self Built Dataset*

<i>Self Built Dataset</i>			
<i>Classification</i>	<i>Recall</i>	<i>Precision</i>	<i>Accuracy</i>
<i>Angry</i>	1.0000	0.8235	0.9842
<i>Disgust</i>	0.2353	0.8889	0.9288
<i>Fear</i>	0.8696	0.8333	0.9815
<i>Happy</i>	1.0000	0.9708	0.9894
<i>Sad</i>	0.9348	0.8431	0.9710
<i>Surprise</i>	1.0000	1.0000	1.0000
<i>Neutral</i>	0.9888	0.8980	0.9710
<i>Weighted Average</i>	0.9129	0.9136	0.9773
<i>Overall</i>	0.9129	0.9129	0.9751

Secara analisis masing-masing kelas, kelas *angry*, *fear*, *happy*, *sad*, dan *neutral* memiliki *recall* dan *precision* yang tinggi yang artinya kelas ditangani dengan sangat baik dengan model. Kecuali kelas *disgust*, yaitu memiliki *recall* yang rendah namun *precision* yang tinggi, yang artinya model tidak dapat mendeteksi kelas dengan baik, tetapi sangat dapat dipercaya ketika mendeteksi kelas. Namun apabila dilakukan analisis secara *overall*, *recall* dan *precision* memiliki nilai yang tinggi yaitu sebesar 91,29%, yang artinya kelas ditangani sempurna oleh model.

Sedangkan pada gambar 4.2 dapat dilihat terdapat 553 citra marah terprediksi marah, 42 citra jijik terprediksi jijik, 359 citra takut terprediksi takut, 1532 citra senang terprediksi senang, 624 citra sedih terprediksi sedih, 646 citra terkejut terprediksi terkejut, dan 827 citra netral terprediksi netral. Dari hasil pengujian arsitektur CNN 8 lapisan konvolusi menggunakan *data augmentation* didapatkan akurasi data validasi sebesar 63,85%. Dengan *precision*, *recall*, dan *accuracy* sebagai berikut :

**Tabel 4.14** TPR (*True Positive Rate/Sensitivity/Recall*), PPV (*Positive Predictive Value/Precision*), dan *Accuracy FER2013*

<i>FER2013</i>			
<i>Classification</i>	<i>Recall</i>	<i>Precision</i>	<i>Accuracy</i>
<i>Angry</i>	0.5772	0.5302	0.8753
<i>Disgust</i>	0.3784	0.3784	0.9808
<i>Fear</i>	0.3506	0.4891	0.8551
<i>Happy</i>	0.8636	0.8705	0.9345
<i>Sad</i>	0.5004	0.5311	0.8364
<i>Surprise</i>	0.7774	0.7154	0.9384
<i>Neutral</i>	0.6707	0.5696	0.8564
<i>Weighted Average</i>	0.6385	0.6344	0.8860
<i>Overall</i>	0.6385	0.6385	0.8967

Secara analisis masing-masing kelas, kelas *angry*, *disgust*, *fear*, *sad* memiliki nilai *recall* dan *precision* yang rendah, artinya beberapa kelas tersebut tidak ditangani dengan baik oleh model. Sedangkan kelas *neutral* memiliki *recall* yang tinggi, namun *precision* yang rendah, artinya kelas terdeteksi dengan baik, namun menyertakan titik kelas lain di dalamnya. Sedangkan kelas *Happy* dan *Surprise* memiliki *recall* dan *precision* yang tinggi, artinya model dapat menangani kedua kelas tersebut dengan sempurna pada data validasi FER2013. Namun secara *overall*, *recall* dan *precision* memiliki nilai yang lumayan tinggi, artinya secara keseluruhan kelas model dapat menangani seluruh kelas dengan cukup tinggi.

Secara analisis kedua data validasi (FER2013) dan data uji (*Self Built Dataset*) adalah model yang dibangun dengan 8 layer konvolusi menggunakan *data augmentation* dapat menangani seluruh kelas pada FER2013 dengan cukup baik dan pada *Self Built Dataset* dengan sempurna. Karena masing-masing memiliki nilai akurasi yang didapatkan sebesar 63,85% dan 91,29%.

Berikut adalah perbandingan arsitektur CNN yang dibangun sendiri dengan beberapa arsitektur CNN yang populer digunakan :

**Tabel 4.15** Pengujian arsitektur populer

<i>Arsitektur CNN</i>	<i>Data Augmentation</i>	<i>Train Accuracy</i>	<i>Validation Accuracy</i>	<i>Test Accuracy (Self Built Dataset)</i>
ResNet18	No	0.7671	0.5209	0.4776
DenseNet121	No	0.9163	0.5851	0.6095
MobileNetv2	No	0.4510	0.3756	0.3931
VGG16	No	0.2513	0.2471	0.3509
<b>Self Built CNN</b>	No	0.9796	0.6439	0.6913
ResNet18	Yes	0.5184	0.5038	0.6412
DenseNet121	Yes	0.5833	0.5624	0.6781
MobileNetv2	Yes	0.2513	0.2471	0.3509
VGG16	Yes	0.2513	0.2471	0.3509
<b>Self Built CNN</b>	Yes	0.6786	0.6385	<b>0.9129</b>

Berdasarkan tabel di atas, beberapa arsitektur CNN mengalami *overfitting* pada bagian di mana tidak menggunakan *data augmentation* antara lain ResNet, DenseNet, serta Self Built CNN. Sedangkan untuk arsitektur CNN yang menggunakan *data augmentation*, hasilnya tidak ada yang mengalami *overfitting*. Beberapa arsitektur CNN seperti VGG dan MobileNet yang tidak terlalu kompleks, sehingga menghasilkan nilai akurasi yang rendah.

Untuk hasil akurasi data uji tertinggi pada arsitektur CNN yang populer adalah DenseNet dengan akurasi sebesar 60,85% tanpa *data augmentation* dan 67,81% menggunakan *data augmentation*. Namun berdasarkan hasil-hasil uji coba terhadap beberapa arsitektur CNN populer, Arsitektur CNN yang dibangun sendiri memiliki nilai akurasi data uji yang lebih tinggi daripada akurasi arsitektur CNN populer lainnya yaitu sebesar 69,13% tanpa *data augmentation* dan 91,29% menggunakan *data augmentation*.

## 5. Kesimpulan

Berdasarkan hasil penelitian yang dilakukan, kinerja dari sistem yang dibangun menggunakan *Convolutional Neural Network* berjumlah 8 lapisan konvolusi menggunakan *data Augmentation* dan diuji menggunakan metode *cross dataset* citra orang-orang Indonesia didapat nilai akurasi sebesar 91,29% dengan *sensitivity* atau *recall* atau *True Positive Rate (TPR)* sebesar 91,29%, *precision* atau *Positive Predictive Value (PPV)* sebesar 91,29%, dan *overall accuracy* sebesar 97,51%. Berdasarkan hasil analisis nilai *recall* dan *precision* yang tinggi, artinya kelas pada data uji ditangani dengan sempurna oleh model yang dibangun.

Berdasarkan tujuan penelitian ini yaitu membangun sistem pengenalan ekspresi wajah dari *cross dataset* menggunakan metode *Convolutional Neural Network (CNN)*. Sistem deteksi ekspresi wajah menggunakan data latih non-Indonesia dapat diterapkan pada data uji Indonesia apabila menggunakan pemodelan sistem yang tepat. Pada penelitian ini, model arsitektur *CNN* yang tepat berhasil ditemukan setelah melalui beberapa tahap evaluasi. Penelitian ini memiliki batasan-batasan tertentu yaitu tidak menggunakan deteksi dan hanya memiliki tujuh ekspresi, data uji yang terbatas, serta data latih yang memiliki banyak anomali.

Saran untuk penelitian berikutnya dengan topik yang sama dengan penelitian ini, diperbanyak data uji, memilih dataset yang baik atau tidak banyak anomali sebagai data latih apabila ingin melakukan metode *cross dataset*. Penggunaan deteksi wajah dapat digunakan agar mempermudah pengumpulan *dataset* sehingga tidak perlu melakukan *crop* citra secara manual.

## Referensi

- [1] Y. Huang, F. Chen, S. Lv, and X. Wang, "Facial expression recognition: A survey," *Symmetry (Basel)*, vol. 11, no. 10, 2019, doi: 10.3390/sym11101189.
- [2] M. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba, "Coding facial expressions with Gabor wavelets," *Proc. - 3rd IEEE Int. Conf. Autom. Face Gesture Recognition, FG 1998*, no. May, pp. 200–205, 1998, doi: 10.1109/AFGR.1998.670949.
- [3] R. Breuer and R. Kimmel, "A Deep Learning Perspective on the Origin of Facial Expressions," pp. 1–16, 2017.
- [4] A. Mollahosseini, D. Chan, and M. H. Mahoor, "Going deeper in facial expression recognition using deep neural networks," *2016 IEEE Winter Conf. Appl. Comput. Vision, WACV 2016*, 2016, doi: 10.1109/WACV.2016.7477450.
- [5] J. A. Mehrabian, A.; Russell, "An Approach to Environmental Psychology," *MIT Press Cambridge, MA, USA*, 1974.
- [6] A. Hidaka and T. Kurita, "Consecutive Dimensionality Reduction by Canonical Correlation Analysis for



- Visualization of Convolutional Neural Networks,” *Proc. Isc. Int. Symp. Stoch. Syst. Theory its Appl.*, vol. 2017, no. 0, pp. 160–167, 2017, doi: 10.5687/sss.2017.160.
- [7] H. Abhirawan, Jondri, and A. Arifianto, “Pengenalan Wajah Menggunakan Convolutional Neural Networks (CNN),” *Univ. Telkom*, vol. 4, no. 3, pp. 4907–4916, 2017.

## Lampiran